

A2  
In response to these and other problems, an improved system and method is provided for parsing in a distributed directory-enabled environment using an eXtensible Markup Language ("XML") application program interface. The method accepts an XML file as an input stream, parses the input stream, and scans the input stream for an object. Upon finding an object, the method determines whether the object references a system service and dynamically loads the referenced service. The service is dynamically configured and the object is instantiated in a class factory.

In the claims:

Please cancel claims 2-20 without prejudice or disclaimer.

Please add the following new claims:

0955222.05001  
3/5/01  
A3  
Cm  
21. (New) A method for parsing in a distributed directory-enabled application environment using an eXtensible Markup Language ("XML") application program interface, the interface including a class factory, the method comprising:  
accepting an XML file as an input stream,  
parsing the input stream,  
scanning the input stream for an object,  
determining whether the object references a system service,  
dynamically loading the service if referenced,  
dynamically configuring the service, and  
instantiating the object in the class factory, so that the service referenced by the object in the XML stream is automatically available to the object.

22. (New) The method of claim 21 further including determining whether the service is available before dynamically loading and configuring the service.

23. (New) The method of claim 22 further including determining whether the service is already loaded before loading the service.

24. (New) The method of claim 22 further including:  
determining if the service is available; and

defaulting the object to a document object model during instantiation in the class factory if the service is unavailable.

25. (New) The method of claim 24 further including:  
determining if there is a suitable document object model; and  
defaulting the object to a highest available class during instantiation in the class factory if there is no suitable document object model.

26. (New) The method of claim 21 further including scanning the input stream for a plurality of objects.

27. (New) The method of claim 21 further including accepting a plurality of XML files as the input stream.

28. (New) A computer system for parsing in a distributed directory-enabled application environment using an eXtensible Markup Language ("XML") application program interface, the interface including a class factory, the system comprising:

at least one processor;  
at least one memory accessible to the processor;  
an application stored in a portion of the memory; and  
software for parsing an XML file for the application, the software comprising instructions  
for:

accepting the XML file as an input stream,  
parsing the input stream,  
scanning the input stream for an object,  
determining whether the object references a system service,  
dynamically loading the service if referenced,  
dynamically configuring the service, and  
instantiating the object in the class factory, so that the service referenced by the  
object in the XML stream is automatically available to the object..

29. (New) The system of claim 28, wherein the software further includes instructions  
for:

B  
A3  
cm  
10986601  
20030101

determining whether the service is available before dynamically configuring and loading the service.

30. (New) The system of claim 29, wherein the software further includes instructions for:  
determining whether the service is already loaded before loading the service.

31. (New) The system of claim 29, wherein the software further includes instructions for:  
determining if the service is available; and  
defaulting the object to a document object model during instantiation in the class factory if the service is unavailable.

32. (New) The system of claim 31, wherein the software further includes instructions for:  
determining if there is no suitable document object model; and  
defaulting the object to a highest available class during instantiation in the class factory if there is no suitable document object model.

33. (New) The system of claim 28, wherein the software further includes instructions for:  
scanning the input stream for a plurality of objects.

34. (New) The system of claim 28, wherein the software further includes instructions for:  
accepting a plurality of XML files as the input stream.

In the Abstract:

Please delete the Abstract paragraph in its entirety and add the following:

An improved system and method is provided for parsing in a distributed directory-enabled environment using an eXtensible Markup Language ("XML") application program interface. The method accepts an XML file as an input stream, parses the input stream, and

B1  
A4  
Cm.t

FOIA b 7 - D

A4  
Cm.t